Decisões de Projeto - Vumbora

1. Foi ultilizado o Padrão Singleton porque assim só seria necessário referenciar o objeto uma vez; o Singleton foi utilizado nas classes de Sistemas (SistemaCarona, SistemaDeBairros, SistemaNotificacao, SistemaUsuarioCRUD, SistemaUsuarioLogin, SistemaUsuariosDefault).
2. Foram criados diferentes sistemas para diminuir o acoplamento e dividir as responsabilidades do sistema de caronas como um todo.
3. Foi optada uma divisão de classes modularizadas em um sistema, facilitando as alterações que inclui no conceito de simplicidade, usabilidade e extensabilidade.
4. Foi escolhida a estrutura de dados LinkedList na classe Carona por sua facilidade em inserir e remover elementos.
5. Foram utilizadas diferentes Controllers para deixar mais simples e clara a integração entre o back e o front end, de forma que uma controller conhece sua definição e o que outra controller deixa.
6. Foi decidido unificar o usuário para que este possa ser motorista e passageiro ao mesmo tempo, tal mudança pode ser notada se comparada a primeira parte do Vumbora.
7. Foi decidido utilizar o Log do próprio Play Framework por este ser mais simples e cabível ao propósito do projeto.
8. Foi decidido que o log seria encapsulado e podendo ser visto em AdminLogger para diminuir acomplamento com outras classes e faclitar a leitura do código.. Nesta classe as ações são registradas e salvas.
9. Foram utilizadas classes Enum para simplificar as mensagens default para o sistema. Sendo mais rápido e legível para o programador e para o usuário.
10. Cada Controller tem seu próprio Logger para facilitar o debug, visto como boa prática de programação.
11. O Padrão Strategy foi utilizado na busca, para promover a extensabilidade da ação. O programador poderá implementar diferentes tipos de busca, se assim desejar.
12. Para uma melhor tratamento de String em busca por bairro foi criada um sistema (SistemaDeBairros) que contém uma lista com todos os bairros de Campina Grande e, além disso, métodos para a criação e remoção de bairros.
13. Foi decidido que cada notificação (do tipo Notificação) que contém um tipo (Pedido, Cancelamento, Aceitação ou Rejeição)e assim, cada usuário armazena duas listas de notificações (lidas e não lidas) para facilitar a pesquisa de cada notificação.